Amendment to the Claims:

This listing of claims replaces all prior versions, and listings, of claims in the application:

Kindly cancel claims 9-17, amend the claims as follows, and substitute new claims 25-27 for the cancelled claims.

---

1. (Currently amended) A method of aligning instructions in a processor comprising:

B2

storing a plurality of instructions of different sizes in a plurality of buffer areas, each buffer area storing a unit instruction width;

aligning a first instruction from said buffer areas;

decoding ~~the~~ a size of the first instruction;

selecting at least one of said buffer areas to output said first instruction on an output part;

during said outputting, first determin~~ing the~~ a beginning of a second instruction based on the size of the first instruction, ~~,~~ decoding the size of the second instruction~~,~~, and second determining whether processing the second instruction will deplete ~~one of a~~ said plurality of buffers; and

based on said second determining, instructing ~~the one of~~ the plurality of buffers to receive additional ~~data if~~

~~processing the second instruction depletes the one of the plurality of buffers~~ instructions.

B2

2. (Cancelled)

3. (Original) The method of Claim 1, further comprising comparing a most significant bit of a pointer to a first of the plurality of sub-buffers to a most significant bit of a pointer to a second of the plurality of sub-buffers to determine whether processing one of the plurality of instructions will deplete a buffer.

4. (Original) The method of Claim 1, further comprising storing a first instruction across a plurality of storage elements prior to processing the instructions.

5. (Original) The method of Claim 1, further comprising adding the size of the first instruction to a current instruction position to determine the beginning of the second instruction.

6. (Original) The method of Claim 1, further comprising aligning ahead a number of cycles equal to a cache latency.

B2

7. (Original) The method of Claim 1, further comprising aligning instructions in a digital signal processor.

8. (Original) The method of Claim 1, further comprising issuing a request to a memory to reload the plurality of buffers.

9-17. (Cancelled)

18. (Currently amended) The processor of Claim ~~16~~ 27, wherein the transition detector compares a most significant bit of a pointer to a first of the plurality of sub-buffers to a most significant bit of a pointer of a second of the plurality of sub-buffers to determine whether processing one of the plurality of instructions will deplete a buffer.

19. (Currently amended) The processor of Claim ~~16~~ 27, wherein the processor aligns ahead a number of cycles equal to a cache latency.

20. (Currently amended) The processor of Claim ~~16~~ 21, wherein the processor is a digital signal processor.

B2

21. (Currently amended) An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to align instructions in a processor, the instructions causing the machine to:

storing a plurality of instructions of different sizes in a plurality of buffer areas, each buffer area storing a unit instruction width;

~~receive data containing instructions in a plurality of buffers;~~

decode ~~the~~ a size of a first instruction from said buffer areas;

select at least one of said buffer areas to output said first instruction on an output part;

during said outputting, first determine ~~the~~ a beginning of a second instruction based on the size of the first instruction~~;~~, decode the size of the second instruction~~;~~, and second determine whether processing the second instruction will deplete ~~one of the~~ said plurality of buffers; and

based on said second determine, instruct ~~the one of~~ the plurality of buffers to receive additional ~~data if processing the second instruction depletes the one of the plurality of buffers~~ instructions.

22. (Original) The apparatus of Claim 21, wherein the plurality of instructions are stored in a plurality of sub-buffers.

23. (Original) The apparatus of Claim 21, wherein a most significant bit of a pointer to a first of the plurality of sub-buffers is compared to a most significant bit of pointer to a second of the plurality of sub-buffers to determine whether processing one of the plurality of instructions will deplete a buffer.

24. (Original) The apparatus of Claim 21, wherein a first instruction is stored across a plurality of storage elements prior to processing the instructions.

25. (New) A method of processing instructions within a processor, comprising:

storing instructions of different widths within a cache, having a plurality of subportions, each subportion in the cache storing a unit instruction width, where an instruction of unit width takes up a single subportion in the cache, and an

instruction of more than said unit width takes up more than one subportions within the cache;

multiplexing each of the portions of said cache to an output point, and selecting one of said cache portions as a current instruction;

during said selecting said current instruction, predicting which of said instructions within said cache will be depleted of instruction data within a number of cycles approximately equal to a latency of the cache, and instructing loading of that number of buffers with additional instruction information.

26. (New) A method as in claim 25, wherein said predicting comprises comparing a most significant bit of a pointer to a first subportion, to a most significant bit of a pointer to a second subportion, to determine if any of the subportions will be depleted.

27. (New) A processor comprising:

a plurality of buffer areas, adapted to store a plurality of instructions of different width in a plurality of subparts, each of said subparts storing a unit instruction width, and said instructions of greater than unit instruction widths being stored in multiple said subparts;

a multiplexer, connected to said plurality of subparts, and selecting and aligning one of said plurality of subparts from any of said subparts within said buffer areas as a current instruction; and

B2

a predictor, operating to predict when at least one of the plurality of buffer areas will be empty, and to send a signal to instruct said at least one of the plurality of buffer areas to load another instruction data.